

Lock接口. (比synchronized更有扩展性)

表5-1 Lock接口提供的synchronized关键字不具备的主要特性

| 特 性 | 描 述 |
|-----------|----------------------------------------------------------------|
| 尝试非阻塞地获取锁 | 当前线程尝试获取锁，如果这一时刻锁没有被其他线程获取到，则成功获取并持有锁 |
| 能被中断地获取锁 | 与 synchronized 不同，获取到锁的线程能够响应中断，当获取到锁的线程被中断时，中断异常将会被抛出，同时锁会被释放 |
| 超时获取锁 | 在指定的截止时间之前获取锁，如果截止时间到了仍旧无法获取锁，则返回 |

队列同步器 AQS

- ①定义：用来构建锁或其他同步组件的基础框架。
采用state表示同步状态。
采用FIFO队列完成资源获取线程的排队工作。

② 队列同步器。

表5-3 同步器可重写的方法

| 方法名称 | 描 述 |
|---------------------------------------------|-------------------------------------------------------|
| protected boolean tryAcquire(int arg) | 独占式获取同步状态，实现该方法需要查询当前状态并判断同步状态是否符合预期，然后再进行 CAS 设置同步状态 |
| protected boolean tryRelease(int arg) | 独占式释放同步状态，等待获取同步状态的线程将有机会获取同步状态 |
| (续) | |
| 方法名称 | 描 述 |
| protected int tryAcquireShared(int arg) | 共享式获取同步状态，返回大于等于 0 的值，表示获取成功，反之，获取失败 |
| protected boolean tryReleaseShared(int arg) | 共享式释放同步状态 |
| protected boolean isHeldExclusively() | 当前同步器是否在独占模式下被线程占用，一般该方法表示是否被当前线程所独占 |

- 模板方法
- ①独占式。→ 同一时刻只能有一个线程获取到锁。
 - ②共享式。 } 获取与释放同步状态。
 - ③查询同步队列中的等待线程。

分析了独占式同步状态获取和释放过程后，适当做个总结：在获取同步状态时，同步器维护一个同步队列，获取状态失败的线程都会被加入到队列中并在队列中进行自旋；移出队列（或停止自旋）的条件是前驱节点为头节点且成功获取了同步状态。在释放同步状态时，同步器调用tryRelease(int arg)方法释放同步状态，然后唤醒头节点的后继节点。

重入锁。

定义：支持一个线程对资源的重复加锁。

定义：支持一个线程对资源的重复加锁。

支持公平、非公平锁。

默认，可以极大提高吞吐量。

① 实现重入。
要解决 { 线程再次获取锁。
锁的最终释放。

读写锁 提升读的并发程度。

① 读写状态设计。一个整型变量维护多个读线程和一个写线程的状态。
按位切割使用。

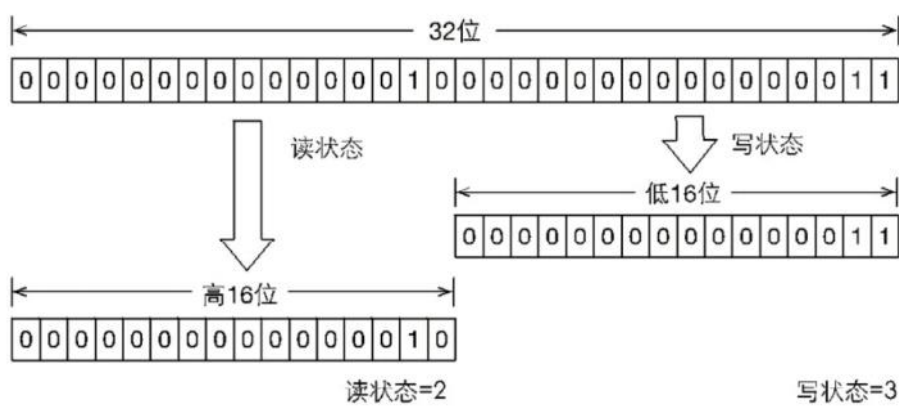


图5-8 读写锁状态的划分方式

② 锁降级 (为保证数据可见性)

线程A获取写锁，再获取读锁，再释放写锁。

LockSupport工具。

提供线程阻塞和唤醒功能。

带对象的版本可以打印阻塞对象的信息。

Condition接口。配合Lock接口，实现一个同步队列，多个等待队列。

表5-12 Object的监视器方法与Condition接口的对比

| 对比项 | Object Monitor Methods | Condition |
|----------------------------|-------------------------|--------------------------------------------------------------|
| 前置条件 | 获取对象的锁 | 调用 Lock.lock() 获取锁 调用 Lock.newCondition() 获取 Condition 对象 |
| 调用方式 | 直接调用 如：object.wait() | 直接调用 如：condition.await() |
| 等待队列个数 | 一个 | 多个 |
| 当前线程释放锁并进入等待状态 | 支持 | 支持 |
| 当前线程释放锁并进入等待状态，在等待状态中不响应中断 | 不支持 | 支持 |
| 当前线程释放锁并进入超时等待状态 | 支持 | 支持 |
| 当前线程释放锁并进入等待状态到将来的某个时间 | 不支持 | 支持 |
| 唤醒等待队列中的一个线程 | 支持 | 支持 |

| 二三四五六七八九十 | 又 | 又 |
|------------------------|-----|----|
| 当前线程释放锁并进入等待状态到将来的某个时间 | 不支持 | 支持 |
| 唤醒等待队列中的一个线程 | 支持 | 支持 |
| 唤醒等待队列中的全部线程 | 支持 | 支持 |

Java中的锁

2021年12月23日 19:18